

# Dental caries classification using depthwise separable convolutional neural network for teledentistry system

Trie Maya Kadarina<sup>1</sup>, Zendi Iklima<sup>1</sup>, Rinto Priambodo<sup>2</sup>, Riandini<sup>3</sup>, Rika Novita Wardhani<sup>3</sup>

<sup>1</sup>Department of Electrical Engineering, Faculty of Engineering, Universitas Mercu Buana, Jakarta, Indonesia

<sup>2</sup>Department of Information System, Faculty of Computer Science, Universitas Mercu Buana, Jakarta, Indonesia

<sup>3</sup>Department of Electrical Engineering, Politeknik Negeri Jakarta, Depok, Indonesia

## Article Info

### Article history:

Received Jul 17, 2022

Revised Sep 30, 2022

Accepted Oct 18, 2022

### Keywords:

Convolutional neural network

Dental caries

Depthwise separable convolution

Image classification

Teledentistry

## ABSTRACT

Caries may be halted or reversed in their progression by early detection, better hygiene habits, and coadministered drugs. The major clinical procedures for identifying dental caries are visual-tactile examination and dental radiography. However, due to their location, approximate caries exceedingly difficult to detect and affect the clinical assessment. Incorrect interpretations may also hinder the diagnostic procedure. Computational approaches and technology can be used to help dentists assess caries. Teledentistry has the ability to improve dental health care by providing access to dental care services from a remote location. Teledentistry helps identifying various stages of caries lesions using neural network and devices connected to the internet. This research develops an image classification for teledentistry systems using depthwise separable convolutional neural network. The trainable parameters reduction of depthwise separable convolution (DSC) successfully reduces the computational cost of conventional convolutional neural networks (CNN). As a result, the DSC model is reduced by 91.49% when compared to the traditional CNN model. Several DSC models improve conventional CNN accuracies in the training, validation, evaluation, and testing stages.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## Corresponding Author:

Zendi Iklima

Department of Electrical Engineering, Faculty of Engineering, Universitas Mercu Buana

South Meruya, Kembangan, Jakarta, Indonesia

Email: zendi.iklima@mercubuana.ac.id

## 1. INTRODUCTION

Teledentistry has enormous potential to enhance oral health service by expanding access to the telemedicine domain through the use of digital technology, communication technology, and dentistry [1]. Compared to traditional clinical treatments, teledentistry procedures can exhibit comparable efficacy and cost [2]. Teledentistry is a branch of telemedicine that offers convenience in dental health services such as diagnosis, action plans, consultations, and follow-up via digital transmission, which may be accessible from anywhere. Teledentistry may have a substantial impact on dental health care in rural places, for example, by giving consultation advice and supporting services in remote areas, thus saving travel expenses, waiting time, and unproductive time [3]. With teledentistry, a dentist can diagnose the various stages of a caries lesion using photos of the teeth taken from a smartphone or an intraoral camera. The findings show that severe lesions and healthy teeth may both be detected with high sensitivity; for example, the detection specificities for lesions at all stages were always higher than 83.3% [4].

Caries is the consequence of a continual process of multiple demineralization and remineralization cycles. Improved hygiene habits, early detection of caries, and supportive therapy may contain the potential

to halt or reverse the progression of the condition. Clinically, visual-tactile examination and dental radiography are the primary methods for detecting dental caries [5]. However, due to their location, caries approximations are extremely difficult to recognize, making clinical examination challenging. Furthermore, inaccurate interpretations may obstruct the diagnostic procedure. Computational method and technologies can be employed to assist dentists in caries assessment [6].

The Laplacian filtering, adaptive threshold, statistical feature extraction, and back-propagation neural network are components of a computational algorithm which is widely used as a dental caries diagnostic system. The system shows that the back-propagation neural network can effectively classify a tooth surface as having caries or not having caries. The back-propagation neural network was trained over 105 intra-oral digital radiographies, which were annotated by the dentist. The model was performed with a learning rate of 0.4, momentum of 0.2, and 500 iterations. The model has 97.1% accuracy [7].

A graphics and intelligence-based script technology (GIST) descriptor was implemented to extract important information from dental caries images for black's categorization. Decision trees, fuzzy sugeno, neural networks, K-nearest neighbor, AdaBoost, and Naive Bayes are approached to categorize the primary and reduced characteristics. According to the findings, the AdaBoost classifier has 90.92% sensitivity and 90% specificity when used to diagnose infected teeth [8]. A modified linearly adaptive particle swarm optimization (LA-PSO) demonstrates a great binary classification of dental caries and non-caries. The feature extraction uses a grey level co-occurrence matrix (GLCM), which is acquired from the individual panoramic X-ray image. The combination of LA-PSO and GLCM yielded 99% accuracy [9].

Orthopantomogram (OPG) and radiovisiography (RVG) are two types of X-ray detectors used in dentistry. The upper and lower teeth are captured in a single picture by OPG. RVG takes X-ray pictures that are utilized to diagnose a specific tooth. Both detectors provide X-ray pictures, which are the most commonly used techniques for diagnosing dental disorders. In medical image processing, deep learning algorithms have a huge range of uses, including the classification of dental caries. A transfer learning architecture, namely VGG16, was trained using 251 (1,000×1,496 pixels) OPG and RVG X-ray dental images. The findings demonstrate the excellent accuracy of 88.46% with which the transfer learning architecture can identify OPG and RVG X-ray dental images [10]. Deep convolutional neural networks (DCNNs) have yielded impressive results in the classification of dental caries on periapical radiographs. By distributing 3,000 periapical radiographic images, 80% of which were training set images and 20% of which were validation set images, the DCNN's diagnostic showed the accuracies of premolar, molar, and both premolar and molar models were 89.0%, 88.0%, and 82.0% [11].

In a different study, a classification technique that combines convolutional neural networks (CNN) and long short-term memory (LSTM) model was developed for use in periapical dental images for the detection and diagnosis of dental caries. The CNN-LSTM model was trained using 1,500 dental X-ray images in 300 iterations. According to the results, an optimal CNN-LSTM model performed with 96% accuracy [12]. 112 bitewing radiographs were augmented to improve CNN to classify three labels, namely normal, incipient, and advanced. The augmentation result produces 3,464 images. The transfer learning models were selected, namely Inception and ReNet, which were trained in 11,500 iterations. The result shows the highest area under the curve (AUC) value of 86.1% [6]. Based on the faster-regional CNN (Faster-RCNN), an attempt was made to devise a robust approach for detecting and classifying various oral and dental disorders using OPG pictures. The technology will give a revolutionary approach for detecting and classifying different types of teeth (incisors, molars, premolars, and canine teeth), as well as some underlying oral oddities including fixed partial dentures and impacted teeth. The results show that Faster-RCNN has a detection accuracy of 91.03% [13]. The CNN were also trained on the annotated data to achieve panoramic X-ray segmentation. The dataset containing 1,000 images is divided into 14 classes, with each class representing a distinct dental issue. Over 200 epochs, an efficient residual factorized ConvNet (ERFNet) was performed, yielding 98 % accuracy, 98% precision, 91% recall, and 93% F1-score [14].

A modified DCNN has been studied to classify dental cavities from open-source datasets, namely the Kaggle dataset that is classified into cavities and non-cavities. The modified DCNN model was trained using 74 images and 30 iterations. With binary cross entropy loss and a learning rate of 0.001, the DCNN model was built for the classification of dental caries and non-caries. The DCNN model was able to obtain a maximum accuracy of 71.43% by adjusting hyperparameters [15]. By increasing the dataset images, a transfer learning model was trained using 500 images containing caries and non-caries labels. In 100 iterations, VGG16, VGG19, InceptionV3, and ResNet50 were successfully trained with 99.37%, 98.48%, 99.89%, and 98.01% accuracy, respectively [16].

This research proposed a classifier to support the teledentistry system shown in Figure 1 to identify the image that was captured using the intraoral camera. The controller will transform the input images to the classifier as tensors. The result will be transmitted to the web server for storing and displaying on the front-end as a smart teledentistry using the deep learning method. Therefore, this research proposed a new modified model which simplifies the conventional CNN architecture, namely the depthwise separable

convolution (DSC) neural network. Our proposed model is designed to support a teledentistry system that integrates with DSC. The convolution layer of a traditional CNN is optimized using DSC in order to save computational costs by lowering the number of trainable parameters while maintaining similar results.

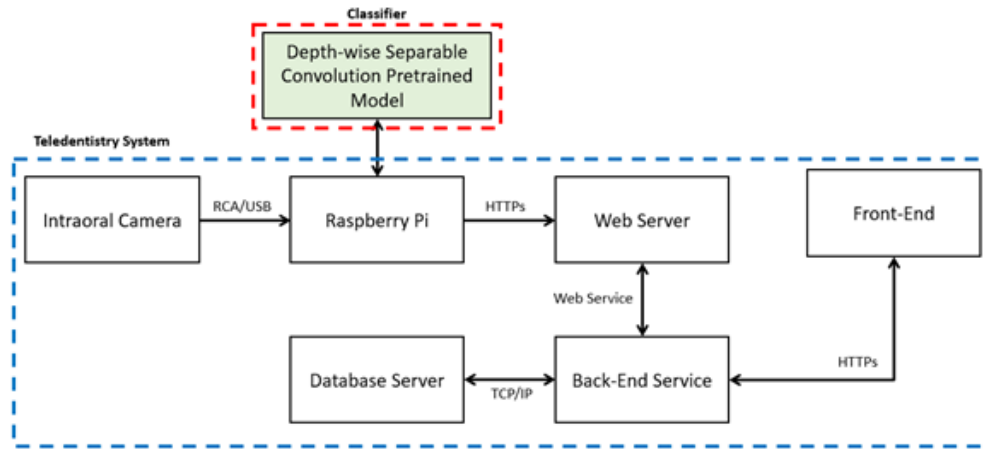


Figure 1. Teledentistry system on clinic

## 2. METHOD

### 2.1. Datasets

To ensure that the model's performance remains stable, the dataset is divided into 10% random testing distribution and 90% training distribution. The training distribution is also divided into 67% as the training set, 23% as the validation set, and 10% as the evaluation set. Figure 2 shows the dental caries dataset.

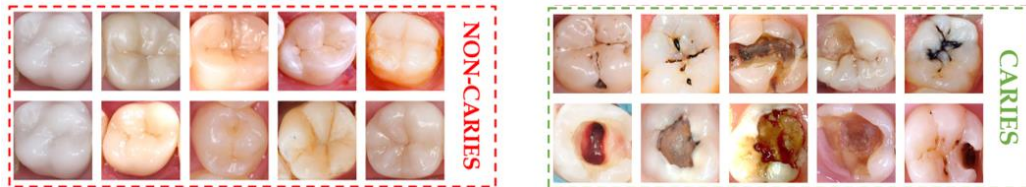


Figure 2. Dental caries dataset

The dataset was enhanced with ImageDataGenerator, which offers standardization, rotation, shifts, flips, rescale, and other features. The name ImageDataGenerator refers to real-time data augmentation, which implies that the model receives fresh variants of the pictures while using less memory. The ImageDataGenerator configured 23% of the validation set along random flips, with a rotation range of 2 degrees, a zoom range of 0.1, and a rescale of 1/255.

### 2.2. Depthwise separable convolutional neural network

CNN is commonly utilized as a classifier for image classification. CNN reduced image dimension using a downsample technique, namely feature extraction. The output of the feature extraction is flattened as the classifier input [17], [18]. Figure 3 shows a conventional CNN architecture. The output of feature extraction can be formulated as [19]:

$$y_{w^{l+1}, h^{l+1}, d} = \sum_{w=0}^W \sum_{h=0}^H \sum_{d=0}^D f_{w,h,d} \times x_{h^{l+1}, w^{l+1}, d}^l \quad (1)$$

by assuming  $(x_{i^{l+1}}^l)$  as an input tensor which comprises of triple indexes including height  $(h^l)$ , width  $(w^l)$ , and depth  $(d^l)$ . Spatial location of  $(h^l, w^l)$  used from bank filter of  $f$  and  $d^l$  is a receptive field in  $x^l$ . Therefore, the total trainable parameters of the feature extraction represent as Kernel formulated as [20]:

$$y_{k^{l+1},c^{l+1}} = D_k^2 \times D_c^2 \times M \times N \quad (2)$$

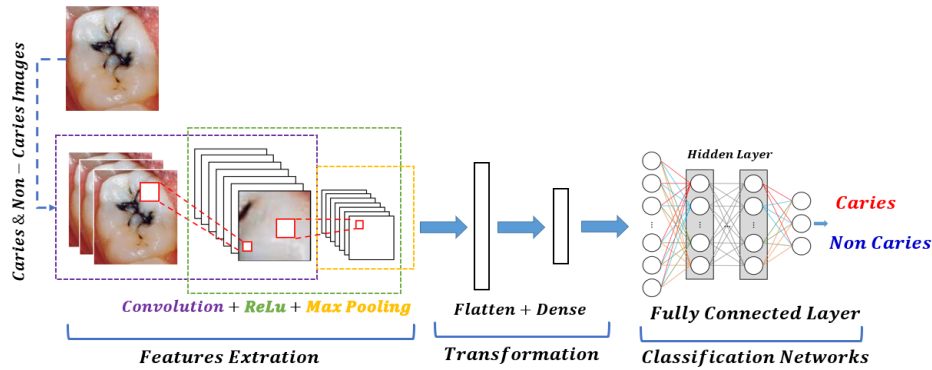


Figure 3. Conventional CNN architecture [15]–[18], [21]

Considering how many convolutional parameters there are in the Kernel that need to be computed as vectors, CNN models with high-resolution images demand additional memory allocation. As a result, by decreasing convolutional trainable parameters, numerous CNN models may be simplified. Using DSC, the convolution layer of a conventional CNN is optimized to reduce computational costs by lowering the number of trainable parameters while maintaining equivalent results [22], [23]. The downsample architecture of DSC is represented in Figure 4.

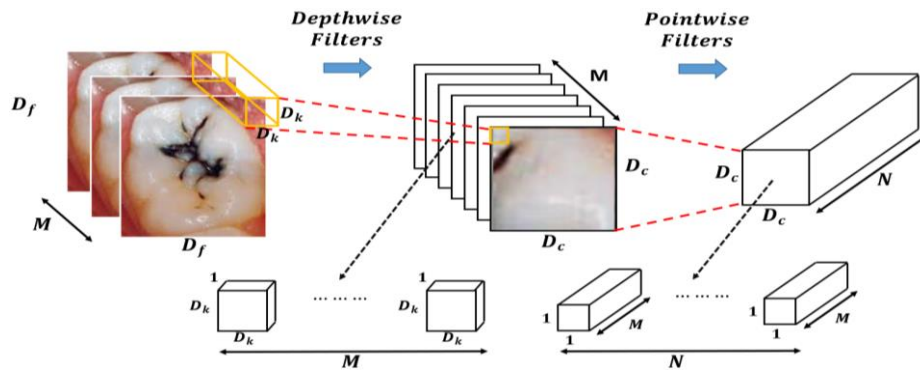


Figure 4. DSC downsample architecture

Based on Figure 4, DSC consists of a depthwise filter and a pointwise filter, which are formulized as:

$$y_{w^{l+1},h^{l+1},d} = \sum_{d=0}^D f_d \times \sum_{h=0}^H \sum_{w=0}^W f_{w,h} \times x_{h^{l+1}+h,w^{l+1}+w}^l \quad (3)$$

where  $f_d$  denotes as  $1 \times 1$  convolution namely pointwise filter. Therefore, the total trainable parameter of DSC layer formulized as [24]:

$$y'_{k^{l+1},c^{l+1}} = (D_c^2 \times M)(D_k^2 + N) \quad (4)$$

Compares to (2) and (4), DSC shows lower trainable parameters than conventional CNN, which is formulized as [20], [24]. Based on (5), it is proven that DSC reduces the trainable parameter of the convolution layer.

$$\frac{y'_{k^{l+1},c^{l+1}}}{y_{k^{l+1},c^{l+1}}} = \frac{(D_k^2 + N)}{(D_k^2 \times N)} \quad (5)$$

### 3. RESULTS AND DISCUSSION

#### 3.1. Hyperparameters

This research proposed a depth-wise separable convolution architecture of the conventional CNN architecture. As an initial hyperparameter of the training process, the image size was set at  $224 \times 224$  resolution, 100 iterations, and 0.5 dropout. The augmentation setup was utilized to avoid overfitting models. The augmentation setup contains 1/255 of rescale, 2 degrees of rotation range, 0.1 of zoom range, horizontal flip, and vertical flip. The first proposed architecture contains 3 convolutional layers with a total number of filters of 32, 32, and 64, respectively. This layer uses 3 Kernels and relu as an activation function. The maxpooling layer was constructed using a  $2 \times 2$  pool size and placed in front of the convolution layer. The fully connected layer was designed in 2 layers, where the latest layer represents the desired image labels. Figure 5(a) depicts the first proposed conventional CNN architecture, while Figure 5(b) depicts the DSC architecture. The second proposed architecture contains 4 convolution layers with a number of filters of 32, Kernel size (5, 5, 1, and 5), and strides ( $4 \times 4$ ,  $1 \times 1$ ,  $1 \times 1$ , and  $4 \times 4$ ). The fully connected layer uses the same setup as the first proposed architecture. The second proposed conventional CNN architecture is shown in Figure 6(a), whereas the DSC architecture is shown in Figure 6(b).

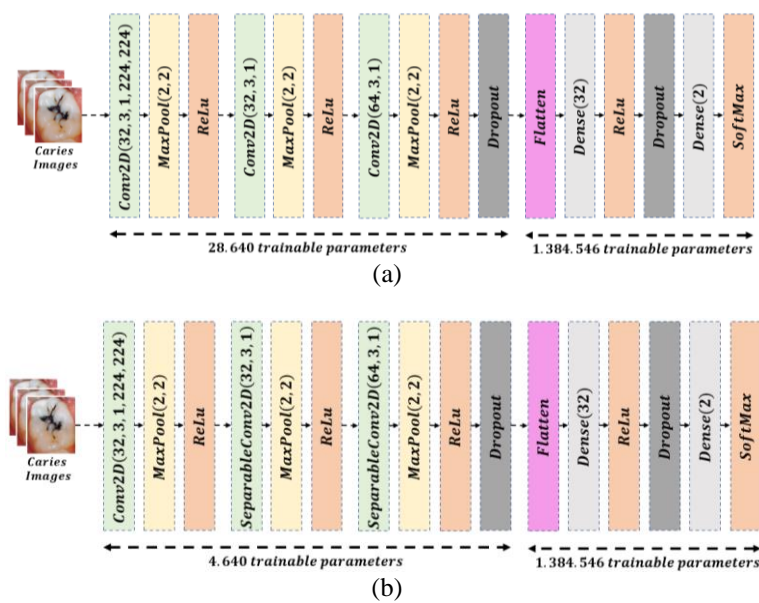


Figure 5. The first proposed architecture of (a) conventional CNN and (b) DSC

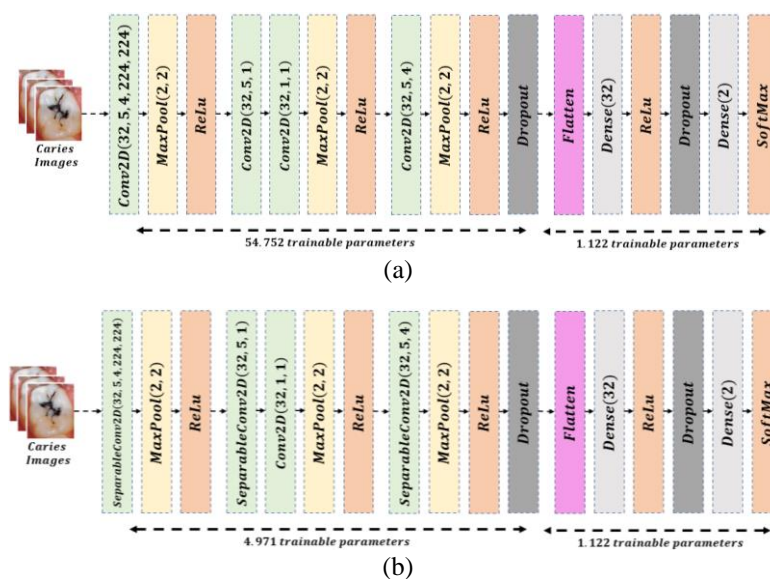


Figure 6. The second proposed architecture of (a) conventional CNN and (b) DSC

Based on Figure 5, the DSC was reduced to 83.79% of the trainable parameters of the first proposed architecture, and in Figure 6, the DSC was reduced to 90.92% of the trainable parameters of the second proposed architecture. The fully connected layer has similar trainable parameters to each proposed architecture. The proposed models then compared to the models that have architectures as shown in Table 1. Odd number models (except model-11) are conventional CNN models, whereas even number models (except model-12) are DSC models from odd number models. Sequentially, model-2 is a DSC model derived from model-1, model-4 is a DSC model derived from model-3, model-6 is a DSC model derived from model-5, and so on. Model-11 is a transfer learning model, namely InceptionV3, and model-12 is a transfer learning model, namely VGG16.

Table 1. Models summary

Models	Hyper. Params.
Model-1 [15]	Conv2D (32, 2, (224, 224, 3)); MaxPooling2D (2, 2); Conv2D (64, 2); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (128); Dropout (0.5); Dense (2);
Model-2	SeparableConv2D (32, 2, (224, 224, 3)); MaxPooling2D (2, 2); SeparableConv2D (64, 2); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (128); Dropout (0.5); Dense (2);
Model-3 [25]	Conv2D (32, 5, (4,4), (224, 224, 3)); Conv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); Conv2D (32, 5, (4,4)); Conv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); Conv2D (32, 5, (4,4)); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-4	Conv2D (32, 5, (4,4), (224, 224, 3)); SeparableConv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); SeparableConv2D (32, 5, (4,4)); SeparableConv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); SeparableConv2D (32, 5, (4,4)); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-5	Conv2D (32, 3, (224, 224, 3)); MaxPooling2D (2, 2); Conv2D (32, 3); MaxPooling2D (2, 2); Conv2D (64, 3); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-6	SeparableConv2D (32, 3, (224, 224, 3)); MaxPooling2D (2, 2); SeparableConv2D (32, 3); MaxPooling2D (2, 2); SeparableConv2D (64, 3); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-7	Conv2D (32, 5, (4,4), (224, 224, 3)); MaxPooling2D (2, 2); Conv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); MaxPooling2D (2, 2); Conv2D (32, 5, (4,4)); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-8	SeparableConv2D (32, 5, (4,4), (224, 224, 3)); MaxPooling2D (2, 2); SeparableConv2D (32, 5, (1,1)); Conv2D (32, 1, (1,1)); MaxPooling2D (2, 2); SeparableConv2D (32, 5, (4,4)); MaxPooling2D (2, 2); Dropout (0.5); Flatten (); Dense (32); Dropout (0.5); Dense (2);
Model-9 [26]	Conv2D (128, 2, (224, 224, 3)); Conv2D (128, 2); MaxPooling2D (2, 2); Conv2D (128, 2); Conv2D (128, 2); MaxPooling2D (2, 2); Conv2D (128, 2); Conv2D (128, 2); MaxPooling2D (2, 2); Flatten (); Dense (1024); Dropout (0.5); Dense (2);
Model-10	Conv2D (128, 2, (224, 224, 3)); SeparableConv2D (128, 2); MaxPooling2D (2, 2); SeparableConv2D (128, 2); Conv2D (128, 2); MaxPooling2D (2, 2); SeparableConv2D (128, 2); Conv2D (128, 2); MaxPooling2D (2, 2); Flatten (); Dense (1024); Dropout (0.5); Dense (2);
Model-11 [16]	inception_v3(299, 299,3); Flatten (); Dense (256); Dropout (0.5); Dense (2);
Model-12	vgg16(224, 224,3); Flatten (); Dense (256); Dropout (0.5); Dense (2);
Model-13 [21]	Conv2D (16, 6, (224, 224, 3)); MaxPooling2D (2, 2); Conv2D (16, 7); MaxPooling2D (2, 2); Conv2D (16, 7); MaxPooling2D (2, 2); Flatten (); Dense (100); Dense (2);
Model-14	SeparableConv2D (16, 6, (224, 224, 3)); MaxPooling2D (2, 2); SeparableConv2D (16, 7); MaxPooling2D (2, 2); SeparableConv2D (16, 7); MaxPooling2D (2, 2); Flatten (); Dense (100); Dense (2);
Model-15 [27]	Conv2D (96, 11, (4,4), (224, 224, 3)); MaxPooling2D (3,3); Conv2D (512, 3, (2,2)); Conv2D (1024, 3); Conv2D (512, 3); MaxPooling2D (3,3); Flatten (); Dense (4096); Dense (4096); Dense (2);
Model-16	Conv2D (96, 11, (4,4), (224, 224, 3)); MaxPooling2D (3,3); SeparableConv2D (512, 3, (2,2)); Conv2D (1024, 3); SeparableConv2D (512, 3); MaxPooling2D (3,3); Flatten (); Dense (4096); Dense (4096); Dense (2);



### 3.2. Model performance

The models were run on an Intel (R) Core i5-6300 HQ CPU running at 2.30 GHz, 16 GB RAM, and a Nvidia GeForce GTX 960 M with 4 GB VRAM. The classifiers were trained using data distribution, which includes training split, validation split, and evaluation split. The percentage of the data distribution was explained in the dataset part. The model performance is shown in Table 2.

Table 2. Model performance

Models	Train. Params.	Tr. Acc (%)	Val. Acc (%)	Eval. Acc (%)	Tr. Loss (%)	Precision (%)		Recall (%)		F1-score (%)	
						N	C	N	C	N	C
Model-1 [15]	49.57 K	87.83	82.08	89.04	65.05	81	97	97	79	88	87
Model-2	49.56 K	95.29	90.01	94.52	14.28	92	100	100	92	96	96
Model-3 [25]	108.19 K	78.61	76.58	90.41	43.16	100	90	89	100	94	95
Model-4	13.09 K	80.98	77.98	94.52	41.84	100	90	89	100	94	95
Model-5	14.13 M	81.08	81.24	95.89	43.49	97	95	94	97	96	96
Model-6	13.89 M	90.84	88.38	94.52	20.47	94	95	94	95	94	95
Model-7	55.87 K	76.36	82.80	97.26	43.72	95	100	100	95	97	97
Model-8	6.093 K	84.73	85.90	98.63	32.24	97	100	100	97	99	99
Model-9 [26]	3.09 M	91.86	88.18	95.89	19.29	97	97	97	97	97	97
Model-10	2.93 M	83.97	81.82	98.63	31.35	95	100	100	95	97	97
Model-11 [16]	256.77 K	80.66	78.92	82.19	44.41	85	85	83	87	84	86
Model-12	256.77 K	74.31	76.58	86.30	62.31	79	97	97	76	87	85
Model-13 [21]	801.56 K	83.89	76.12	90.41	19.89	95	100	100	95	97	97
Model-14	776.98 K	94.16	89.54	95.89	15.33	94	95	94	95	94	95
Model-15 [27]	45.58 M	50.42	46.00	52.05	72.20	0	52	0	100	0	68
Model-16	41.01 M	90.03	86.76	100	22.43	100	100	100	100	100	100

Based on Table 2, the term of the optimization of DSC is only to reduce the computational costs which are shown in the trainable parameter column. Trainable parameter reduction can be calculated by the parameter rates in the feature extraction layer. The reduction of the DSC layer in model-2 against model-1 is 72.58%. The reduction of the DSC layer in model-4 against model-3 is 88.82%. The reduction of the DSC layer in model-6 against model-5 is 83.80%. The reduction of the DSC layer in model-8 against model-7 is 90.92%. The reduction of the DSC layer in model-10 against model-9 is 48.44%. The reduction of the DSC layer in model-14 against model-13 is 91.49%. The reduction of the DSC in model-16 against model-15 is 45.55%. Meanwhile, the DSC model accuracy can be maintained as if it were conventional CNN accuracy. Figure 7(a) represents the highest performance graph of the training accuracies. The highest performance graph of the training losses is depicted in Figure 7(b).

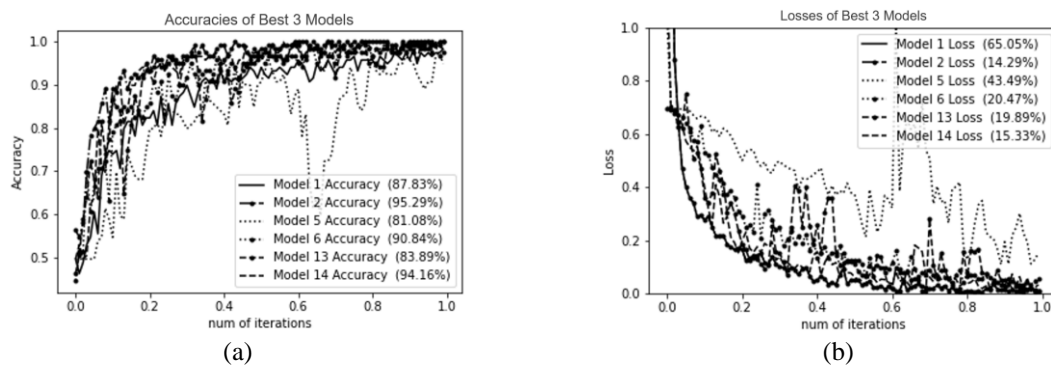


Figure 7. Performance of the best three models (a) accuracies and (b) losses

The proportion of correct classifications was the primary performance variable. The secondary performance indicator was the visualization of CNN-focused features into related image regions. [26]. Based on Table 1, the confusion matrices can be utilized to visualize the consistency of the model performance in test data distribution, which means the data was never trained. We computed the precision, recall, and F1-score for the data distribution of normal labels (N) and carries labels (C). True positives (TP) are accurately predicted positive values that show both the actual and predicted class values are true. Indicating that both the actual and predicted class values are false, true negatives (TN) are correctly predicted negative

values. When the actual class is false but the predicted class is true, this is known as a false positive (FP). Situations where the actual class is true but the predicted class is negative are known as false negatives (FN). The proportion of correctly predicted positive observations to all predicted positive observations is known as precision. The ratio of correctly predicted positive observations to all observations in the actual class is known as recall (sensitivity/true positive rate). The F1 score is calculated by averaging precision and recall. As a result, this score takes into account both false positives and false negatives. The confusion matrices can be formulated as [26]:

$$precision = \frac{TP}{TP+FP} \quad (6)$$

$$recall = \frac{TP}{TP+FN} \quad (7)$$

$$F1 - score = \frac{2 \times precision \times recall}{precision + recall} \quad (8)$$

Figures 8(a)-(p) represents the confusion matrices visualization of model-1 until model-16. Figure 8 represents the visualization of confusion matrices. It refers to the best performance of the proposed DSC as a classifier for the caries problem, which is shown in model-8 and model-16.

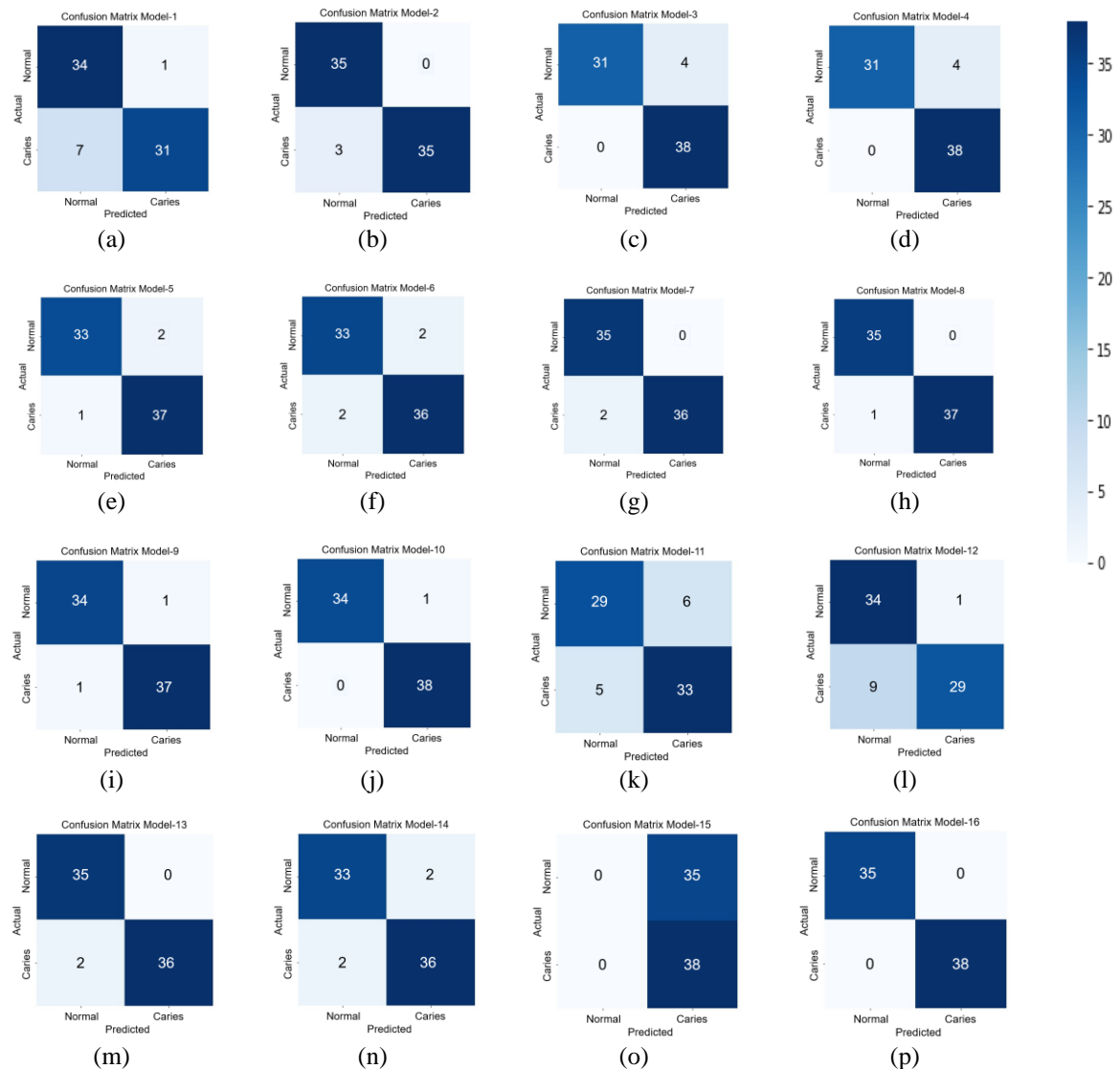


Figure 8. Confusion matrices of (a) model 1, (b) model 2, (c) model 3, (d) model 4, (e) model 5, (f) model 6, (g) model 7, (h) model 8, (i) model 9, (j) model 10, (k) model 11, (l) model 12, (m) model 13, (n) model 14, (o) model-15, and (p) model-16



#### 4. CONCLUSION

In this study, the computational costs were successfully reduced by optimizing the feature extraction layers with a DSC optimization. The DSC reduced the trainable parameters of the conventional CNN. Even with fewer trainable parameters, model accuracy can be maintained at levels comparable to conventional CNN accuracy. The Sonavane model's DSC reduced trainable parameters to 72.58% while successfully improving model accuracy from 86.50% to 94.39%. In addition, the proposed method is compared to transfer learning models such as InceptionV3 and VGG16. Confusion matrices can be used to visualize the consistency of model performance in test data distribution, implying that the data was never trained. We determined the precision, recall, and f1-score for the normal label data distributions (N) and carries label data distributions (C). The primary performance variable was the percentage of correct classifications. The secondary performance variable was the visualization of CNN-focused features in related image regions. The DSC architecture in model-16 outperforms the conventional CNN in model-15. It's improved the whole confusion matrices variable containing the feature extraction reduction, accuracy, recall, precision, F1-score, and execution time.

#### ACKNOWLEDGEMENTS

The authors express gratitude to the Department of Electrical Engineering and Research Center of Universitas Mercu Buana and also to the Department of Electrical Engineering, Politeknik Negeri Jakarta. With their support, the authors have completed this research.

#### REFERENCES




- [1] M. Irving, R. Stewart, H. Spallek, and A. Blinkhorn, "Using teledentistry in clinical practice as an enabler to improve access to clinical care: A qualitative systematic review," *Journal of Telemedicine and Telecare*, vol. 24, no. 3, pp. 129–146, 2018, doi: 10.1177/1357633X16686776.
- [2] S. J. Daniel and S. Kumar, "Teledentistry: A key component in access to care," *Journal of Evidence Based Dental Practice*, vol. 14, pp. 201–208, Jun. 2014, doi: 10.1016/j.jebdp.2014.02.008.
- [3] M. Estai, Y. Kanagasingam, M. Tennant, and S. Bunt, "A systematic review of the research evidence for the benefits of teledentistry," *Journal of Telemedicine and Telecare*, vol. 24, no. 3, pp. 147–156, Apr. 2018, doi: 10.1177/1357633X16689433.
- [4] E. K. Kohara, C. G. Abdala, T. F. Novaes, M. M. Braga, A. E. Haddad, and F. M. Mendes, "Is it feasible to use smartphone images to perform telediagnosis of different stages of occlusal caries lesions?," *PloS one*, vol. 13, no. 9, pp. 1–12, 2018, doi: 10.1371/journal.pone.0202116.
- [5] X. Zhang *et al.*, "Development and evaluation of deep learning for screening dental caries from oral photographs," *Oral Diseases*, vol. 28, no. 1, pp. 173–181, Jan. 2022, doi: 10.1111/odi.13735.
- [6] M. Moran, M. Faria, G. Giraldo, L. Bastos, L. Oliveira, and A. Conci, "Classification of approximal caries in bitewing radiographs using convolutional neural networks," *Sensors*, vol. 21, no. 15, pp. 1–12, Jul. 2021, doi: 10.3390/s21155192.
- [7] V. Geetha, K. S. Aprameya, and D. M. Hinduja, "Dental caries diagnosis in digital radiographs using back-propagation neural network," *Health Information Science and Systems*, vol. 8, no. 1, pp. 1–14, Dec. 2020, doi: 10.1007/s13755-019-0096-y.
- [8] P. Singh and P. Sehgal, "Decision support system for black classification of dental images using GIST descriptors," in *Advanced Computing and Intelligent Engineering*, 2020, pp. 343–352, doi: 10.1007/978-981-15-1081-6\_29.
- [9] M. Sornam and M. Prabhakaran, "A new linear adaptive swarm intelligence approach using back propagation neural network for dental caries classification," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Sep. 2017, pp. 2698–2703, doi: 10.1109/ICPCSI.2017.8392208.
- [10] S. A. Prajapati, R. Nagaraj, and S. Mitra, "Classification of dental diseases using CNN and transfer learning," in *2017 5th International Symposium on Computational and Business Intelligence*, Aug. 2017, pp. 70–74, doi: 10.1109/ISCBI.2017.8053547.
- [11] J.-H. Lee, D.-H. Kim, S.-N. Jeong, and S.-H. Choi, "Detection and diagnosis of dental caries using a deep learning-based convolutional neural network algorithm," *Journal of Dentistry*, vol. 77, pp. 106–111, Oct. 2018, doi: 10.1016/j.jdent.2018.07.015.
- [12] P. Singh and P. Sehgal, "G.V Black dental caries classification and preparation technique using optimal CNN-LSTM classifier," *Multimedia Tools and Applications*, vol. 80, no. 4, pp. 5255–5272, Feb. 2021, doi: 10.1007/s11042-020-09891-6.
- [13] A. Laishram and K. Thongam, "Detection and classification of dental pathologies using faster-RCNN in orthopantomogram radiography image," in *2020 7th Int. Conf. on Signal Proc. and Integrated Net.*, Feb. 2020, pp. 423–428, doi: 10.1109/SPIN48934.2020.9071242.
- [14] M. P. Muresan, A. R. Barbur, and S. Nedevski, "Teeth detection and dental problem classification in panoramic X-ray images using deep learning and image processing techniques," in *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, Sep. 2020, pp. 457–463, doi: 10.1109/ICCP51029.2020.9266244.
- [15] A. Sonavane, R. Yadav, and A. Khamparia, "Dental cavity classification of using convolutional neural network," *IOP Conference Series: Materials Science and Engineering*, vol. 1022, no. 1, pp. 1–9, Jan. 2021, doi: 10.1088/1757-899X/1022/1/012116.
- [16] D. Saini, R. Jain, and A. Thakur, "Dental caries early detection using convolutional neural network for tele dentistry," in *2021 7th International Conference on Advanced Computing and Communication Systems*, 2021, pp. 958–963, doi: 10.1109/ICACCS51430.2021.9442001.
- [17] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," *Pattern Analysis and Applications*, vol. 24, no. 3, pp. 1207–1220, 2021, doi: 10.1007/s10044-021-00984-y.
- [18] D. Singh, V. Kumar, Vaishali, and M. Kaur, "Classification of COVID-19 patients from chest CT images using multi-objective differential evolution-based convolutional neural networks," *European Journal of Clinical Microbiology & Infectious Diseases*, vol. 39, no. 7, pp. 1379–1389, Jul. 2020, doi: 10.1007/s10096-020-03901-z.
- [19] N. K. Chowdhury, M. M. Rahman, and M. A. Kabir, "PDCOVNet: A parallel-dilated convolutional neural network architecture for detecting COVID-19 from chest X-ray images," *Health Inf. Science and Sys.*, vol. 8, no. 1, pp. 1–14, 2020, doi: 10.1007/s13755-020-00119-3.
- [20] G. Wang, G. Yuan, T. Li, and M. Lv, "An multi-scale learning network with depthwise separable convolutions," *IPSN Transactions on Computer Vision and Applications*, vol. 10, no. 1, pp. 1–8, Dec. 2018, doi: 10.1186/s41074-018-0047-6.
- [21] J. Choi, H. Eun, and C. Kim, "Boosting proximal dental caries detection via combination of variational methods and convolutional neural network," *Journal of Signal Processing Systems*, vol. 90, no. 1, pp. 87–97, Jan. 2018, doi: 10.1007/s11265-016-1214-6.
- [22] L. Kaiser, A. N. Gomez, and F. Chollet, "Depthwise separable convolutions for neural machine translation," Jun. 2017, [Online].

Available: <http://arxiv.org/abs/1706.03059>.




- [23] A. A. Ardakani, A. R. Kanafi, U. R. Acharya, N. Khadem, and A. Mohammadi, "Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks," *Computers in Biology and Medicine*, vol. 121, pp. 1–9, Jun. 2020, doi: 10.1016/j.compbiomed.2020.103795.
- [24] B. Liu, D. Zou, L. Feng, S. Feng, P. Fu, and J. Li, "An FPGA-based CNN accelerator integrating depthwise separable convolution," *Electronics*, vol. 8, no. 3, pp. 1–18, Mar. 2019, doi: 10.3390/electronics8030281.
- [25] E. Ihsanto, K. Ramli, D. Sudiana, and T. S. Gunawan, "An efficient algorithm for cardiac arrhythmia classification using ensemble of depthwise separable convolutional neural networks," *Applied Sciences*, vol. 10, no. 2, pp. 1–16, Jan. 2020, doi: 10.3390/app10020483.
- [26] S. Sukegawa *et al.*, "Deep neural networks for dental implant system classification," *Biomolecules*, vol. 10, no. 7, pp. 1–13, Jul. 2020, doi: 10.3390/biom10070984.
- [27] A. Imak, A. Celebi, K. Siddique, M. Turkoglu, A. Sengur, and I. Salam, "Dental caries detection using score-based multi-input deep convolutional neural network," *IEEE Access*, vol. 10, pp. 18320–18329, 2022, doi: 10.1109/ACCESS.2022.3150358.

## BIOGRAPHIES OF AUTHORS






**Trie Maya Kadarina**    received her bachelor's degree in Electrical Engineering from Institut Teknologi Nasional in 2001. She received her master's degree in Biomedical Engineering from the Department of Electrical Engineering, Institut Teknologi Bandung, in 2005. Currently, she is a lecturer in the Department of Electrical Engineering at Universitas Mercu Buana. Her research interests include biomedical instrumentation, electronics and control systems, machine learning, and internet of things. She can be contacted at email: [trie.maya@mercubuana.ac.id](mailto:trie.maya@mercubuana.ac.id).






**Zendi Iklima**    received the M.Sc. degree in Software Engineering from Beijing Institute of Technology in 2018. His research interests include robotics, deep learning, and cloud computing. He has an experience as the Chief Technology Officer of Diaspora Connect Indonesia. Indonesian Diaspora Connect is one of the controversial start-up companies in 2018. The idea of connecting Indonesia is to help the Indonesian government find out about citizens who are studying or living abroad which integrates with artificial intelligence. Based on the idea, he achieved 1st place in ALE Hackathon Competition in 2018 which held by Alcatel-Lucent (Jakarta). He can be contacted at email: [zendi.iklima@mercubuana.ac.id](mailto:zendi.iklima@mercubuana.ac.id).






**Rinto Priambodo**    received his bachelor degree in Electrical Engineering from Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 2004 and his master's degree in Information Technology from Universitas Indonesia in 2011. He is currently pursuing the doctoral degree in the Faculty of Computer Science in Universitas Indonesia. His research interest includes information system, business intelligence, and internet of things in healthcare. He can be contacted at email: [rinto.priambodo@mercubuana.ac.id](mailto:rinto.priambodo@mercubuana.ac.id).



**Riandini**    is active as a Lecturer at the Department of Electrical Engineering, Politeknik Negeri Jakarta, Indonesia. She received her M.Sc in Biomedical Engineering with Full Funded the Netherlands Govt. Scholarship (STUNED) from Rijksuniversiteit Groningen (RuG), The Netherlands 2007. Her research interests include image processing, deep learning and biomedical engineering among many other fields. She can be contacted at email: [riandini@elektro.pnj.ac.id](mailto:riandini@elektro.pnj.ac.id).



**Rika Novita Wardhani**    is a Senior Lecturer and a former Head of Program of Instrumentation and Industrial Control, Department of Electrical Engineering, Politeknik Negeri Jakarta, Indonesia. She received her Master of Engineering in System and Control from the Department of Electrical Engineering, ITS, Surabaya, Indonesia in 1999 fully funded by the YPTS organization. Her research interests cover smart sensor and monitoring system, modelling system, neuro-fuzzy and internet of things. She can be contacted at email: [rika.novitawardhani@elektro.pnj.ac.id](mailto:rika.novitawardhani@elektro.pnj.ac.id).